Intelligent Agents

Kyle Usbeck
Computer Science, Drexel University
http://www.cs.drexel.edu/~kfu22

Slides in part by Dr. William C. Regli
Intelligent Building Control Systems: Current Design

- Highly centralized, hierarchical control systems
- Supervisory Control and Data Acquisition Systems (SCADA)
Intelligent Building Control Systems: A New Idea

• Distributed Intelligence
  – Eliminate central authority
  – No single point-of-failure
  – Facilitates small modifications
  – Fault-tolerant
  – Web-based remote control
A Distributed Approach

• Multi-agent System
• One Agent for Each Function
  – Monitoring (Maintenance, Accounting)
  – Control (Direct Use Reduction, System Max.)
• Types of Agents
  – Simple Reflex (if $x$, then $y$)
  – Model/Utility/Goal-based
  – Learning
What is an agent?

• *No precise definition*
  – “Agentness” is in the eye of the beholder

• **Our** definition:
  Software Systems that are:
  – Communicative
  – Autonomous
  – Capable
  – Adaptive
What is an agent?

- An agent is a software component or system that is:
  - Embedded in, and “aware” of, an environment
  - Dynamic in its behaviors (not single I/O mapping)
  - User enabled/steered, but “empowered” to act for user
  - Able to improve its behavior over time

These are desirable properties for software systems
Examples of Agents

• “Any control system can be viewed as an agent”

• Thermostat
  – Sensor(s): detect room temperature
  – Affector(s): heating on/off, cooling on/off

• Autonomous space probes, fly-by-wire aircraft, nuclear reactor control systems, etc.

  • From Michael Wooldridge “Intelligent Agents: The Key Concepts”
Benefits of Agents: Applied

• Speed/Parallelism
  – Each control function can have its own processing

• Reliability/Redundancy
  – Monitoring agents ensure stability

• Modularity
  – Components designed/upgraded separately

• Reasoning
  – Agents can “learn” and improve behaviors
Then and Now – The Architecture Enablers

- Wood
- Brick
- Steel Skeleton Architecture

World's Tallest Skyscrapers

World's Largest Distributed Agent Based Society

- Procedural-Programming
- Object-Oriented
- Agent-Oriented

FORTRAN
C
JAVA
Cougar
Open Agent Architecture
Other Characteristics of Agents

• Inter-Agent communication
  – Negotiate resources
  – Resolve conflicts
  – Distribute tasks
  – Modalities
    • (I.e. Client/Server, p2p, Broadcast/Subscribe, Directories/LDAP/Services, etc)

• Mobile code/Mobile agents
Historical Development of Agent-Based Computing

• **1970-1990:**
  
  *Distributed Artificial Intelligence (DAI)*

• DAI concepts and potential benefits
  
  – Speed/Parallelism
  – Reliability/Redundancy
  – Modularity
  – Reasoning
Speed & Parallelism

• More programs working on the same problem
• Some computations can be done at once
Reliability & Redundancy

• Not affected too much if one part goes down
• Can be decentralized
Modularity

• By design, DAI programs can be designed in pieces
• Effective partitioning of the problem is required
Reasoning

• Each part of a DAI system can have own reasoning abilities
• It is not necessary for the parts to all “think” the same
Reasoning

• \textit{algorithms} + \textit{knowledge} => \textit{actions}

• Algorithms
  – depend on application
  – often needs customization/tuning for domain
    • real time, latency, bandwidth, etc.

• Knowledge in common format

• Agent takes actions to complete task
Techniques for Reasoning

- Rules, Logic Statements, Expert Systems
- State searches
- Planning, Scheduling
- Bayesian Belief Networks
- Neural Networks
- Genetic Algorithms
- etc....

• Applications
  - Optimization
  - Natural Language Processing (NLP)
  - Natural Language Translation (NLT)
  - Data mining
  - Robotics
Perceived Benefits of DAI

• Maintenance

• Reusability
  – Knowledge
    • Common representations reduce need for duplication of work
  – Algorithms
    • Algorithms typically need only minor adjustments for specific domains

• Platform Independence
Agent Frameworks

• Provide for “easy” creation of agents
• Communication Layer
• Messaging Libraries
• Libraries for accessing services
  – Directory
  – Mediation
  – Security
  – Remittance
  – Operations
Generic Agent Shells

Environment

Data Source/ Resource information

Agent task-specific code that defines this capability (services callable by domain-specific code)

Domain-specific code detailing agent behavior

Input(t)

Output(t+1)

Tuning and/or adaptation
Where do agents fit in?

- Economics
- Logic
- Representation
- Wireless
- Internet

- AI

- Communication

- Cooperation

- DAI

- Other AI Areas

AGENTS
The big picture
Simple Agents

- Termites
- Ants
- Flocking Birds

... all examples of emergent behavior in very simple agents.

Examples taken from "The Computational Beauty of Nature" by Gary W. Flake
Example: Termites

• Observation: populations of creatures cooperative create organized structures
  – termites, bees, beavers
    where are the blueprints for the Beavers’ Dam? The beehive?
• Consider termites with simple rules:
  – wander around randomly until you find a woodchip
  – if carrying a woodchip, drop it and keep wandering
  – if not carrying a woodchip, pick it up and wander
• What happens?
Figure 16.1  Termites randomly placing wood chips according to a simple rule produce order
Example: Ants

• Consider an Ant with the following rules
  – Take a step forward
  – If current space is white, paint it black and turn 90° right
  – If current space is black, paint it white and turn 90° left

• Note:
  – Procedure is time reversible, but there are multiple ways to get to a given state
Ant behavior

• after a few steps

Figure 16.2  Eight steps of Langton’s virtual ant, starting from an initially blank grid
Ant behavior

- after about 10,000 steps

**Figure 16.3** A virtual ant building a highway
How about more than one ant?

- Ants can “interfere” and collaborate

Figure 16.4  Two virtual ants reversing one another’s work

Figure from The Computational Beauty of Nature: Computer Explorations of Fractals, Chaos, Complex Systems, and Adaptation. Copyright © 1998-2000 by Gary William Flake. All rights reserved. Permission granted for educational, scholarly, and personal use provided that this notice remains intact and unaltered. No part of this work may be reproduced for commercial purposes without prior written permission from the MIT Press.
Figure 16.5  Two virtual ants can build more complex structures than a single ant acting alone.
Emergent Behavior

• Whole greater than the sum of parts
• Very simple agents with very simple rules ... when used together, can produce impressive and complex results...
Complex Agents

• Instead of simple rules, agents sense the world and reason about actions
• Goal oriented vs. Procedural
• Communication and collaboration
Agents Talking to themselves

• Individual Software Agents are nothing special
• May have reasoning about a specific domain
Agent Communication

- Distinguished by communicating and collaborating
- Other agents with specialized knowledge
Internet as Agent Testbed

- Massive data
- Chaotic
- Dynamic
- Conflicting
- Lots of Agent Aps:
  - information gathering
  - information filtering
- Infrastructure for building agents and agent grids
Example of a Grid: SETI@home

• SETI:
  Search for Extraterrestrial Intelligence
• Users download a program
• Idle/Background processor time crunches radio data
• Creates massive multi-T-flop virtual computer
• http://setiathome.ssl.berkeley.edu/
SETI@home
Agent Communication: Issues

• Different computer systems
• Different computer programs
• Common “language” is needed
• Directory Services
  – Ie: where to get information
• Routing/Addressing
• Bandwidth
Comm Example: Blackboard

- Agents share common memory space for communication
- Memory space does not have to be on the same computer
Comm Example: Broadcast/Subscribe

- Fault tolerant
- Agents replaced easily
- Security is lacking
- Large numbers of agents affect performance
Comm Example: Peer-2-Peer

• Knows where messages are sent
• Easy to introduce security measures
• Hard to work with in open agent systems
• Gnutella, Kazza?
Agents in Real Life

Space Exploration (e.g. Hubble Space Telescope)
Agents in Real Life

Military Planning and Communications (e.g. SWAT)

Machine A

Machine B

Machine C

Machine D

National Assets
Agents in the Future

Communications

Ubiquitous Computing
Agents in the Future

“Unmanning” the front lines

Robotics

Autonomous Vehicles
DARPA Grand Challenge
Thank You

Kyle Usbeck
kfu22@drexel.edu